

Visualización de Sistemas Dinámicos en el Plano Complejo

Gustavo Goñi

Claudio Delrieux

Departamento de Ing. Eléctrica y Computadoras
Universidad Nacional del Sur, Bahía Blanca, ARGENTINA
e-mail: gmgoni@uns.edu.ar, claudio@acm.org

1 Objetivos

En este proyecto estudiamos las diferentes alternativas para la visualización de mapas complejos y sistemas dinámicos discretos en el campo complejo. En particular, se reseña la experiencia de implementación y uso de un sistema para la visualización de transformaciones conformes basado en la deformación de imágenes de entrada. Uno de los puntos importantes de la implementación, entonces, es la construcción de un *parser* de ecuaciones complejas que permita dotar a la herramienta tanto de versatilidad e interactividad, como de eficiencia a la hora de ejecutar las transformaciones.

El sistema implementado es de utilidad para ejemplificar visualmente muchos de los conceptos formales relativos a las funciones de variable compleja, como polos y ceros, multiplicidad, etc. Para ello, además de imágenes de entrada, es importante dotarlo de la capacidad de realizar la transformación de *paletas* bivariadas [2, 5, 6], específicamente construidas para visualizar con facilidad por ejemplo la existencia, posición y multiplicidad de los polos de la transformación.

Además, la herramienta permite, por su interactividad y versatilidad, ejemplificar la existencia y comprender visualmente la topología de los puntos fijos en la iteración de mapas complejos. Esto es así porque su interfase gráfica permite realizar fácilmente iteraciones de las transformaciones, por lo que determinados teoremas y propiedades de los mapeos complejos, como por ejemplo en los conjuntos de Juliá [3, 4] y otros tópicos de la teoría de los conjuntos fractales y los sistemas caóticos, son representados visualmente de manera clara y precisa. Al mismo tiempo, la aplicación como herramienta para el procesamiento de imágenes es interesante de explorar, como se presentará en algunos ejemplos llamativos.

2 Visualización de Funciones Complejas

Un *número complejo* (\mathbb{C}) es aquel que se puede escribir de la forma $z = x + iy$, donde x y y son números reales (\mathbb{R}) e i es la llamada *unidad imaginaria* cuya existencia se postula tal que $i^2 = -1$. Las cantidades reales x y y son denominadas *componente real* de z y *componente imaginaria* de z respectivamente. Uno de los problemas para comprender el comportamiento de las funciones de variable compleja es el de su difícil representación gráfica. En efecto, las funciones reales del tipo $y = f(x)$ se pueden representar fácilmente mediante una

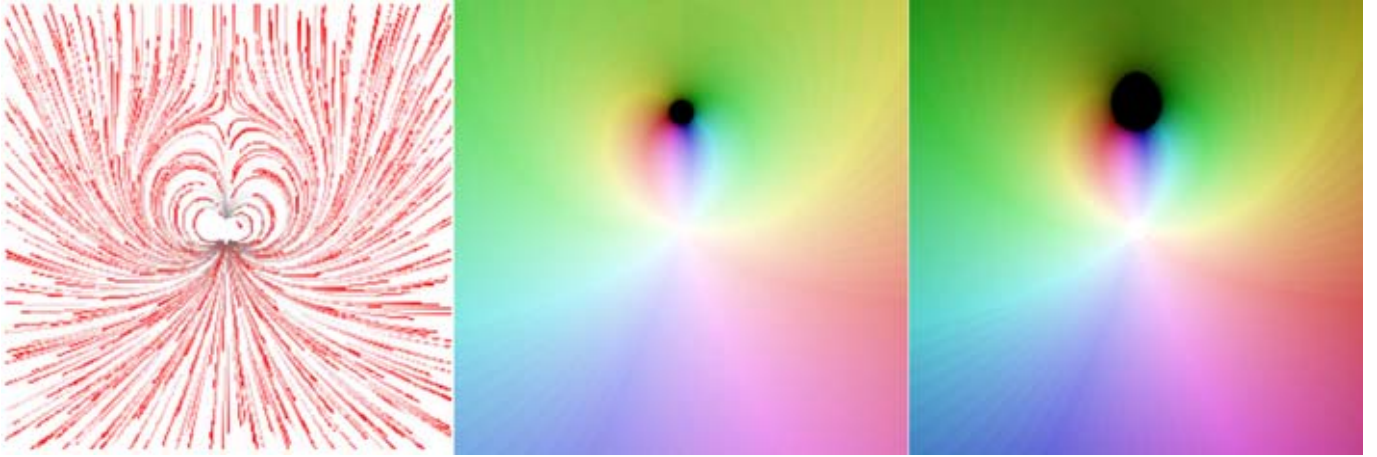


Figura 1: La función $w = \frac{z^2}{z-i}$ representada como mapa vectorial, como mapa cromático y como preimagen distorsionada.

curva (subconjunto de un espacio bidimensional). Si la función es de la forma $z = f(x, y)$ se representará mediante una superficie en un espacio tridimensional. En el dominio complejo, las relaciones funcionales del tipo $w = f(z)$, se pueden escribir como, $u + iv = f(x + iy)$. En este caso intervienen cuatro variables, por lo que se necesitaría un espacio de cuatro dimensiones para representar gráficamente el mapa $w = f(z)$.

De todas maneras existen diversas alternativas para representar gráficamente una función compleja. Una de ellas consiste en realizar un gráfico $\mathbb{R}^2 \rightarrow \mathbb{R}$ considerando alguna propiedad real del valor que entrega la función, por ejemplo: $\Re(w)$, $\Im(w)$, $|w|$ y $\arg(w)$. Con estos datos se puede realizar un gráfico 3D sobre los ejes x, y, z . Otra posibilidad es visualizar al mapeo $w = f(z)$ como un campo vectorial. Esta alternativa es interesante ya que toda la literatura existente en visualización de sistemas dinámicos es directamente aplicable. Este enfoque, sin embargo, no permite visualizar exactamente las propiedades del mapa complejo que pueden resultar de interés. Otra alternativa, propuesta en la Visualización Científica, se basa en representar mapas multidimensionales y/o multivariados por medio de paletas multivariadas. En nuestro caso, podríamos representar el dominio complejo en un plano, y pintar cada punto de dicho dominio con un color que dependa del valor real e imaginario del mapeo en ese punto. Esta alternativa es de mayor utilidad que la anterior, siendo efectiva para visualizar las propiedades de la función compleja bajo estudio.

Para este trabajo investigamos una alternativa diferente para representar estos mapeos, la cual consiste en tomar una imagen de entrada, asociar cada punto de la misma a un valor del dominio (preimagen) de la función compleja, y generar una imagen nueva, en la cual cada punto está pintado con el color correspondiente al punto en su preimagen. Esto establece cierta similitud entre este trabajo y otros sistemas de visualización del comportamiento de sistemas dinámicos basados en imágenes de entrada, como el Line Integral Convolution (LIC) [1]. Esta forma de trabajar, utilizando imágenes de entrada adecuadas, es equivalente a la técnica anterior, aunque, como veremos más abajo, es mucho más versátil. Si elegimos por ejemplo visualizar la función asociando su módulo a la luminancia y su fase a la crominancia (algo adecuado para visualizar polos y ceros), con nuestra técnica sería posible encontrar exactamente la misma visualización que realizando el mapa inverso sobre una imagen en la cual la luminancia disminuye radialmente, y la crominancia cicla angularmente (paleta identidad). En la Fig. 1 mostramos una misma función $w = \frac{z^2}{z-i}$ representada con las tres técnicas mencionadas.

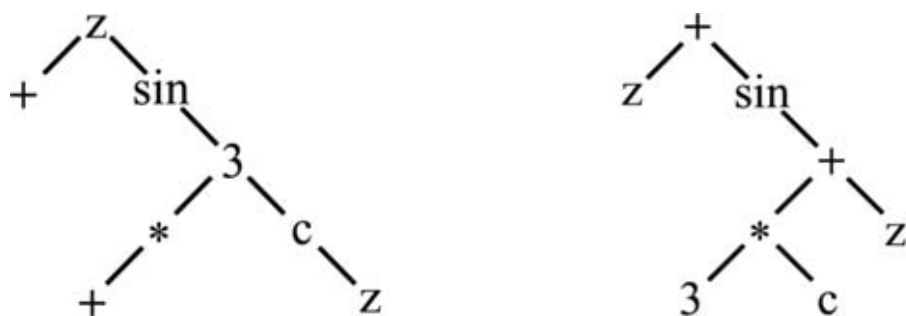


Figura 2: Árboles empleados en el reconocedor sintáctico y evaluador de ecuaciones complejas. $w = z + \sin(3 * c + z)$ (a) Árbol auxiliar (b) Árbol final empleado por el evaluador.

3 Evaluación de Expresiones Complejas

Uno de los elementos indispensables para lograr un sistema interactivo con el usuario, es que el mismo pueda introducir y experimentar con ecuaciones de una manera sencilla y directa. Este objetivo plantea algunas dificultades iniciales que fue necesario resolver. Si bien es posible implementar un sistema que interprete ecuaciones simbólicas en el campo complejo por medio de software matemático (Matlab, Mathematica, Maple), este software, por su misma naturaleza, es inadecuado para producir las visualizaciones en tiempos razonables. La otra alternativa, adoptada en este trabajo, es utilizar un lenguaje de propósito general de adecuada eficiencia, e implementar un reconocedor sintáctico y evaluador para ecuaciones complejas.

El evaluador desarrollado reconoce un conjunto muy variado de funciones (exponenciales, trigonométricas, hiperbólicas, $\Re()$, $\Im()$, módulo, argumento), la constante imaginaria i (usada como prefijo o como sufijo), dos variables (Z y C) y constantes numéricas. El proceso de reconocimiento consta de varias etapas. La primera de ellas es verificar si la ecuación ingresada por el usuario es sintácticamente correcta. A continuación se la manipula transformando las restas en sumas de valores negativos, por lo tanto el signo menos sólo será un operador monario y, de esta forma, también se evita el chequeo de la precedencia en la resta.

El siguiente paso consiste en crear un árbol binario auxiliar que al ser recorrido en modo *in-order* se obtenga el equivalente en notación polaca de la ecuación original. Para ello se diseñó una función recursiva que analiza cada token de la ecuación y que genera una nueva instancia de sí misma cada vez que encuentra un paréntesis izquierdo "(". Cada nodo del árbol creado debe cumplir con las siguientes reglas:

- el nodo ubicado a su izquierda sólo puede contener un operador (+, *, /)
- el nodo ubicado a su derecha sólo puede contener una función, una constante numérica o una variable (z, c, x, y)
- los nodos que contienen operadores sólo pueden apuntar a otro operador, es decir, no deben tener nodos a derecha.

En la última etapa, se contruye un nuevo árbol binario a partir de la ecuación en notación polaca obtenida desde el árbol auxiliar. Éste nuevo árbol es el que utilizaremos para evaluar la ecuación. En la Fig. 2 se muestran a modo de ejemplo los árboles que se obtienen a partir de la ecuación $w = z + \sin(3 * c + z)$.

A la hora de evaluar la ecuación se realiza una recorrida posorden del árbol con un procedimiento recursivo que básicamente actúa de la siguiente forma:

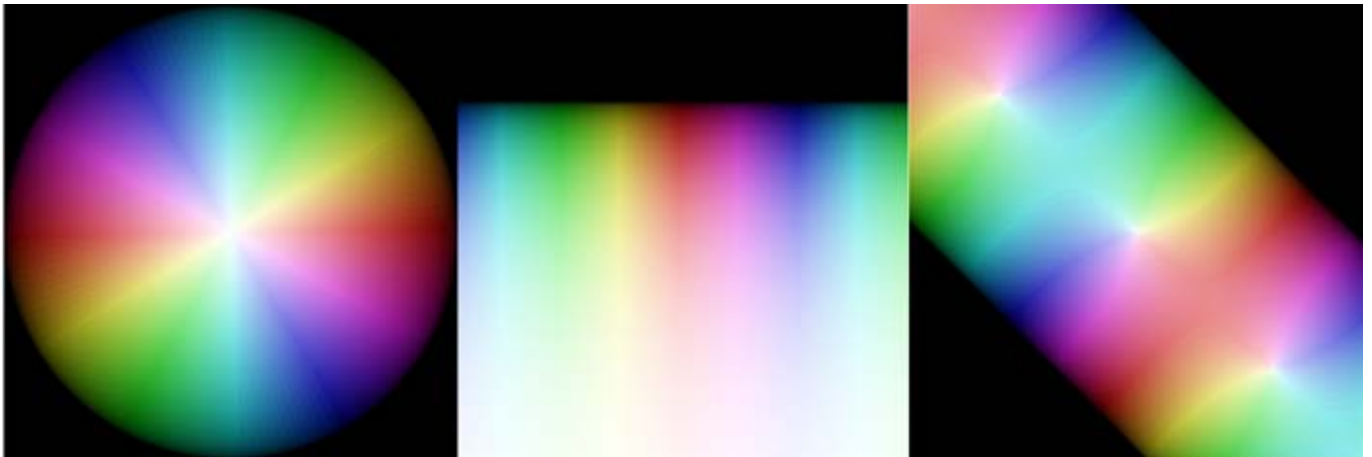


Figura 3: Funciones (a) $w = z^2$, (b) $w = \exp(-iz)$, y (c) $w = \text{sen}((1+i)z)$, representadas como distorsión de una paleta bivariada.

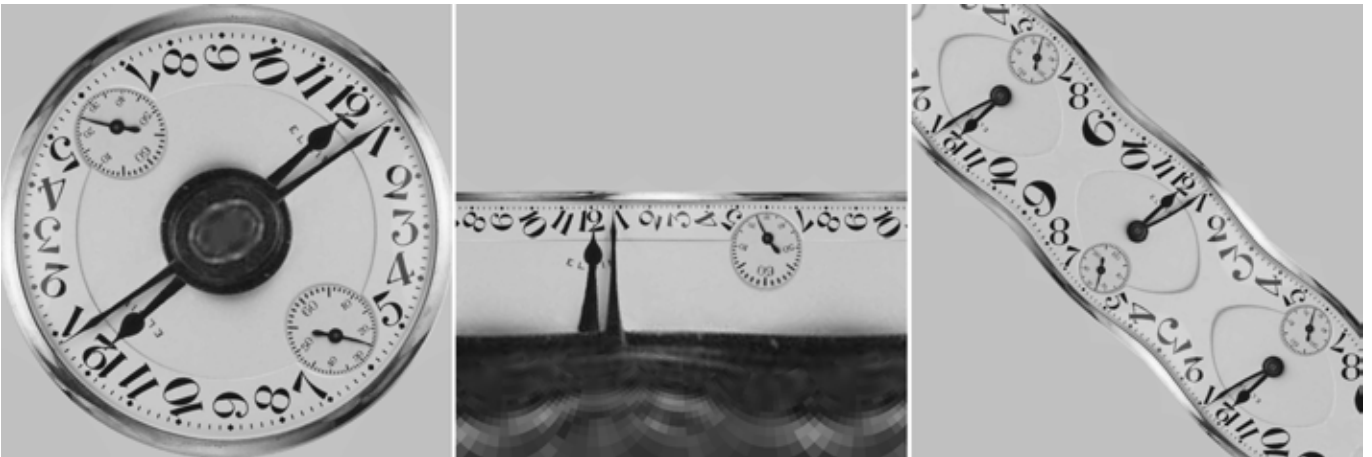


Figura 4: Mapeos de una imagen de entrada.

- Se evalúa el contenido de la rama izquierda del nodo.
- Se evalúa el contenido de la rama derecha del nodo.
- Se aplica la función u operador sobre el resultado obtenido en los paso anteriores.

4 Algunos Ejemplos

En la Fig. 3 mostramos algunos ejemplos de visualización de ecuaciones complejas. Estos resultados son directamente aplicables al procesamiento de imágenes. En efecto, basta solamente con tomar imágenes de entrada arbitrarias (pero adecuadas para nuestro propósito) para producir imágenes de salida que ilustren de una manera inolvidable el resultado de las transformaciones. En la Fig. 4, por ejemplo, mostramos los resultados de las funciones $w = z^2$, $w = \exp(-iz)$ y $w = \text{sen}((1+i)z)$ a una imagen reconocible (un reloj), de manera de entender adecuadamente el comportamiento de la función.

Como ya mencionáramos, el sistema permite iterar los mapeos, de manera de poder rápidamente generar conjuntos fractales a partir de imágenes de entrada (ver Fig. 5). Tam-



Figura 5: Algunas iteraciones de un mapeo complejo, produciendo un punto fijo (conjunto de Juliá).

bién se incluye la posibilidad de modificar las ecuaciones a partir de parámetros, con lo cual se generan animaciones, las cuales son de una notable utilidad para ilustrar la ocurrencia de bifurcaciones y otro tipo de fenómenos.

Referencias

- [1] B. Cabral and L. Leedom. Imaging Vector Fields Using Line Integral Convolution. *ACM Computer Graphics (SIGGRAPH Proceedings)*, 25(3):263–270, 1993.
- [2] Peter Keller and Mary Keller. *Visual Cues: Practical Data Visualization*. IEEE Computer Society Press, Los Vaqueritos, CA, 1990.
- [3] B. Mandelbrot. *The Fractal Geometry of Nature*. W. H. Freeman, New York, 1983.
- [4] H.-O. Peitgen and D. Saupe. *The Science of Fractal Images*. Springer-Verlag, New York, 1986.
- [5] P. K. Robertson and J. O’Callaghan. The Generation of Color Sequences for Univariate and Bivariate Mapping. *IEEE Computer Graphics and Applications*, 6(2), 1986.
- [6] Colin Ware. Color Sequences for Univariate Maps: Theory, Experiments and Principles. *IEEE Computer Graphics and Applications*, 8(5):41–49, 1988.